REMARKS

Claims 1-28 are pending in the application. In the Office Action of September 13, 2004, the Examiner made the following disposition:

A.) Objected to the numbering of claims.

B.) Rejected claims 1-14 and 16-28 under 35 U.S.C. §103(a) as being allegedly unpatentable over *Santhanam (U.S. Patent No. 6,286,135)* in view of *Hamada et al. (U.S. Patent No. 6,493,863)*.

C.) Rejected claim 14 under 35 U.S.C. §103(a) as being allegedly unpatentable over *Santhanam (U.S. Patent No. 6,286,135)* in view of *Hamada et al. (U.S. Patent No. 6,493,863)*, and further in view of *Kahn et al. (U.S. Patent No. 6,662,278)*.

Applicants respectfully traverse the rejections and address the Examiner's disposition below.

A.) Objection to the numbering of claims:

For clarity, the original claims have been canceled without prejudice and reintroduced as correctly numbered, newly added claims 29-56.

Applicants respectfully submit the objection has been overcome and request that it be withdrawn.

B.) Rejection of claims 1-14 and 16-28 under 35 U.S.C. §103(a) as being allegedly unpatentable over *Santhanam (U.S. Patent No. 6,286,135)* in view of *Hamada et al. (U.S. Patent No. 6,493,863):*

Applicants respectfully disagree with the rejection.

Independent claims 29, 36, 44, 51, and 56 each claim subject matter relating to code segments distributed between blocks of memory, or to associating code segments with blocks of memory/memory regions. Claims 29, 36, 44, 51, and 56 are briefly described below.

Independent claim 29 claims a method for developing a data flow program comprising code segments distributed between blocks of memory. A graph representing the data flow program is generated, the graph comprising nodes corresponding to selected ones of the blocks of memory and arcs corresponding to dependency relationships between the nodes.

Similarly, claims 36 and 44 each claim subject matter relating to a data flow program comprising code segments that operate on data in memory. A memory is divided into blocks and each block is associated with at least a portion of the data and with at least one code segment. A graph representation of the data flow program is generated, the graph representation comprising

nodes associated with the blocks, and arcs associated with dependencies between the blocks.

Claims 51 claims subject matter relating to a data flow program and a data flow development tool that generates a graph representation of the data flow program by associating data processed by the data flow program to blocks in the memory, by associating code segments of the data flow program to the blocks, and by determining dependencies between the blocks.

Claim 56 claims a data processing system for developing a data flow program comprising code segments that operate on data in memory. The data processing system comprises means for apportioning a memory into regions and associating the data and the code segments of the data flow program with the regions. The system also comprises means for determining dependencies between the regions. The system further comprises means for generating a graph representation of the data flow program, the representation comprising nodes associated with the blocks, and dependencies between blocks.

Therefore, independent claims 29, 36, 44, 51, and 56 each claim subject matter relating to code segments distributed between blocks of memory, or to associating code segments with blocks of memory/memory regions.

This is clearly unlike *Santhanam* in view of *Hamada*, which fails to disclose or suggest code segments distributed between blocks or memory or associating code segments with blocks of memory/memory regions. *Santhanam* discloses a compiler optimization algorithm that deals with strength reduction of integer machine instructions found in loops (*Santhanam* Abstract). To optimize compilation, *Santhanam* creates a control flow graph that is a low level representation of a procedure. (*Santhanam* 4:10-18). As described in *Santhanam*:

> the nodes of the graph are referred to as basic blocks. These **blocks are sequences of instructions** or low level intermediate operations that are to be executed without a change in the control flow. The edges of the control flow graph would correspond to possible transfers of control between the nodes, depending on conditional checks.

(*Santhanam* 4:19-23)(emphasis added).

Thus, unlike claims 29, 36, 44, 51, and 56, *Santhanam's* blocks are not blocks of memory, but are instead sequences of instructions. For example, as shown in *Santhanam* Figure 17B, *Santhanam* groups instructions into blocks 61 that are arranged in a graph and connected by edges. As shown, each of *Santhanam's* block includes one or more instructions. Thus, *Santhanam's* blocks fail to even relate to blocks of memory. Instead, *Santhanam's* blocks are

merely groups of instructions, with teaching of a relationship to memory regions.

The Examiner argues that *Santhanam* Figures 18-26 teach dividing a memory area into blocks and associating each block with at least one code segment, however, Applicants respectfully disagree. Figures 18-26 and the related text in *Santhanam* merely describes how to handle partitioned addresses during compilation. As discussed in *Santhanam*, an address can be partitioned into a segment specifier and a segment offset. (*Santhanam* 21:58-22:9). When compiling 32-bit programs in a 64-bit address space, *Santhanam* teaches how to translate 32-bit partitioned addresses for the 64-bit address space. (*Santhanam* 21:10-33). Contrary to the Examiner's assertions, Figures 18-26 and the associated text fail to even relate to dividing memory areas into blocks and associating each block with at least one code segment.

Therefore, *Santhanam* fails to disclose or suggest claims 29, 36, 44, 51, and 56.

Further, *Hamada* also fails to disclose or suggest code segments distributed between blocks or memory or associating code segments with blocks of memory/memory regions. *Hamada* discloses creating graphs having nodes that represent processes. (*Hamada* 7:9). Hardware resources HW (*e.g.*, adders and multipliers) that correspond to the processes in the nodes are assigned to the nodes. (*Hamada* 7:16-23). There is no discussion or suggestion in *Hamada* that its nodes are blocks of memory. Thus, *Santhanam* in view of *Hamada* still fails to disclose or suggest claims 29, 36, 44, 51, and 56.

Claims 30-35, 37-43, 45-50 and 52-55 depend directly or indirectly from claims 29, 36, 44 or 51 and are therefore allowable for at least the same reasons that claims 29, 36, 44 and 51 are allowable.

Applicants respectfully submit the rejection has been overcome and request that it be withdrawn.

C.) Rejection of claim 14 under 35 U.S.C. §103(a) as being allegedly unpatentable over *Santhanam (U.S. Patent No. 6,286,135)* in view of *Hamada et al. (U.S. Patent No. 6,493,863)*, and further in view of *Kahn et al. (U.S. Patent No. 6,662,278)*:

Applicants respectfully disagree with the rejection.

Applicant's independent claim 36 is allowable over *Santhanam* in view of *Hamada* as discussed above. *Kahn* also fails to disclose or suggest code segments distributed between blocks or memory or associating code segments with blocks of memory/memory regions. Therefore, *Santhanam* in view of *Hamada* and further in view of *Kahn* still fails to disclose or suggest claim 36.

Claim 42 depends directly or indirectly from claim 36 and is therefore allowable for at least the same reasons that claim 36 is allowable.

Applicants respectfully submit the rejection has been overcome and request that it be withdrawn.

CONCLUSION

In view of the foregoing, it is submitted that claims 29-56 are patentable. It is therefore submitted that the application is in condition for allowance. Notice to that effect is respectfully requested.

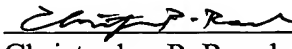
Respectfully submitted,

 (Reg. No. 45,034)

Christopher P. Rauch
SONNENSCHEIN, NATH & ROSENTHAL LLP
P.O. Box #061080
Wacker Drive Station - Sears Tower
Chicago, IL 60606-1080
Telephone 312/876-2606
Customer #26263
Attorneys for Applicant(s)

CERTIFICATE OF MAILING

I hereby certify that this correspondence is being deposited as First Class Mail in an envelope addressed to Commissioner for Patents, PO Box 1450, Alexandria, Virginia 22313-1450 on March 11, 2005.

 (Reg. No. 45,034)

Christopher P. Rauch